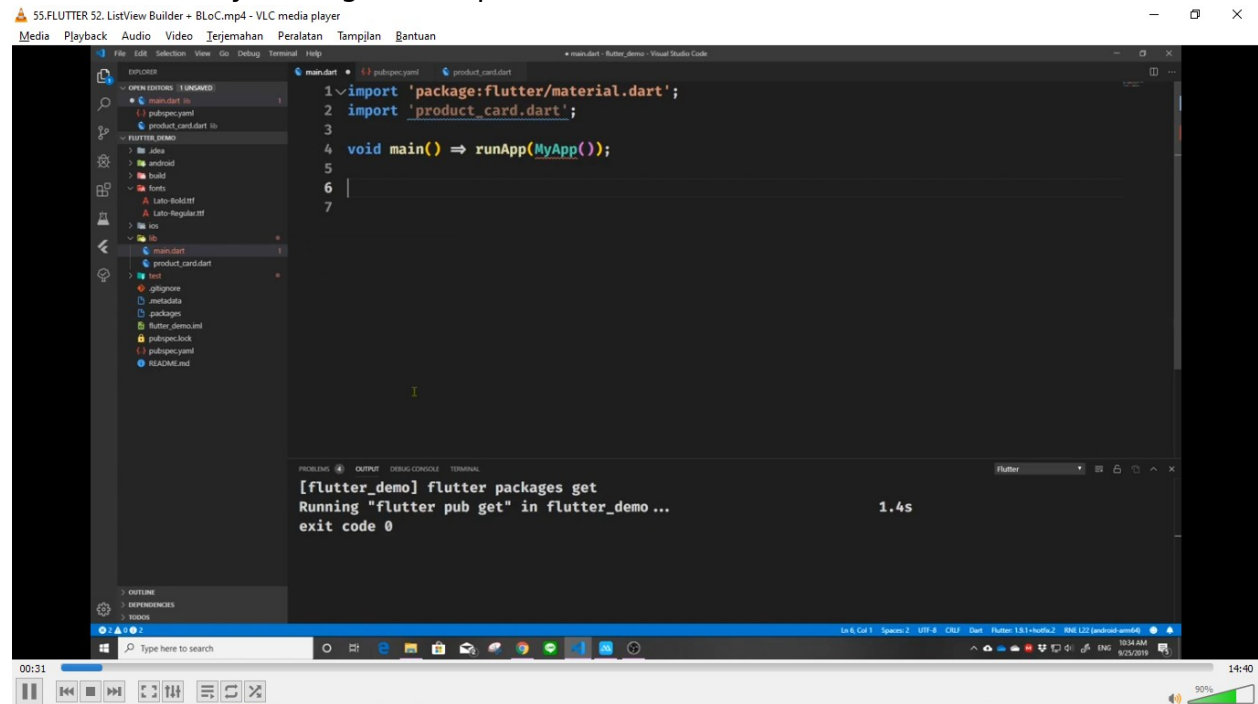
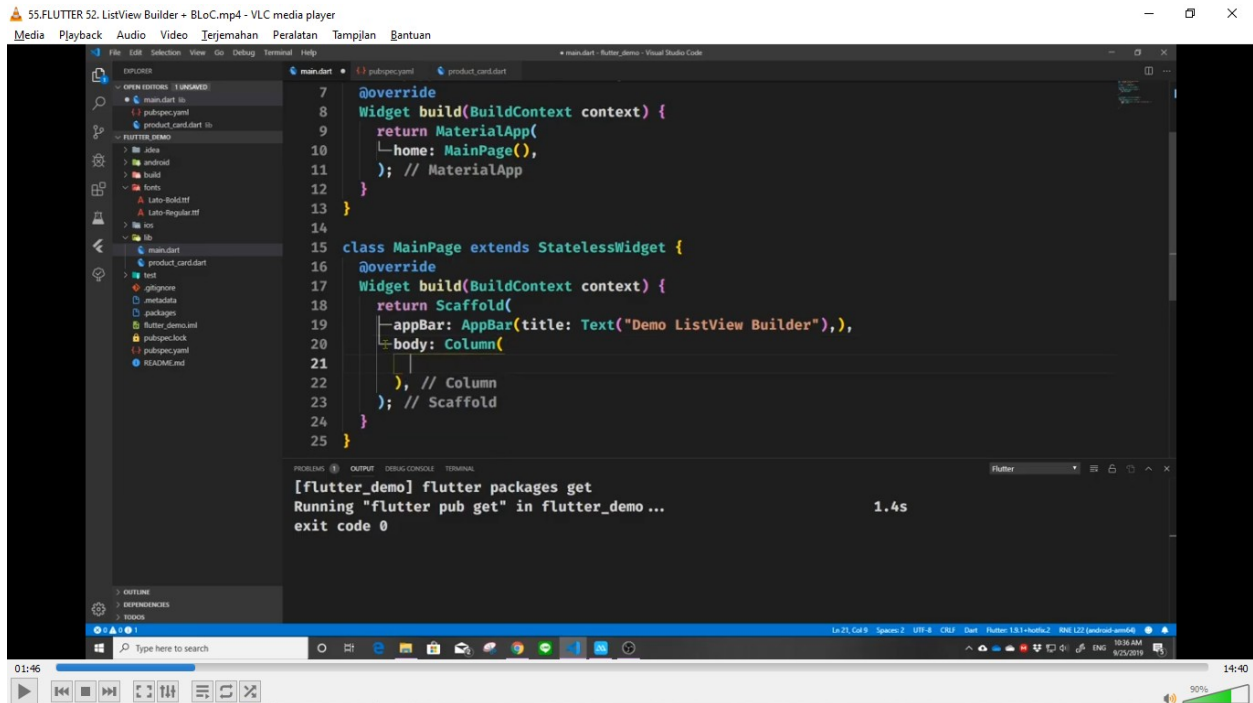


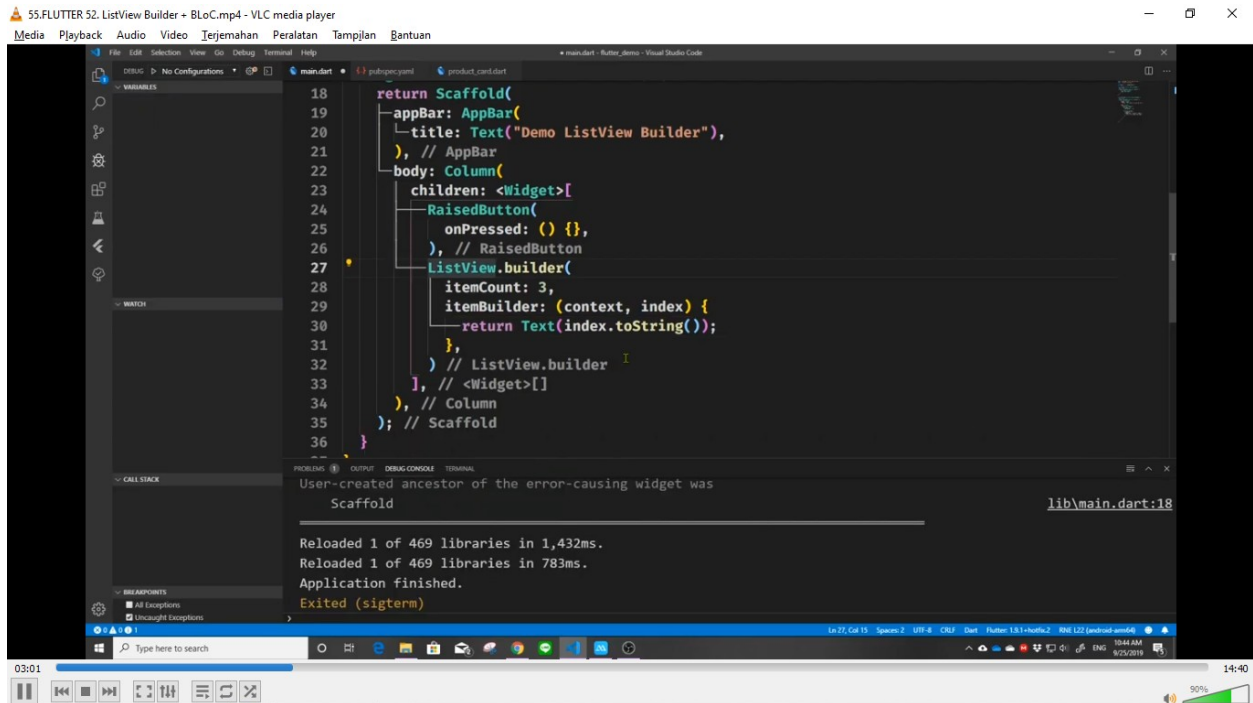
Menjelaskan menggunakan listview builder
Listview builder gampang banget
Sebelum itu saya menggunakan product card



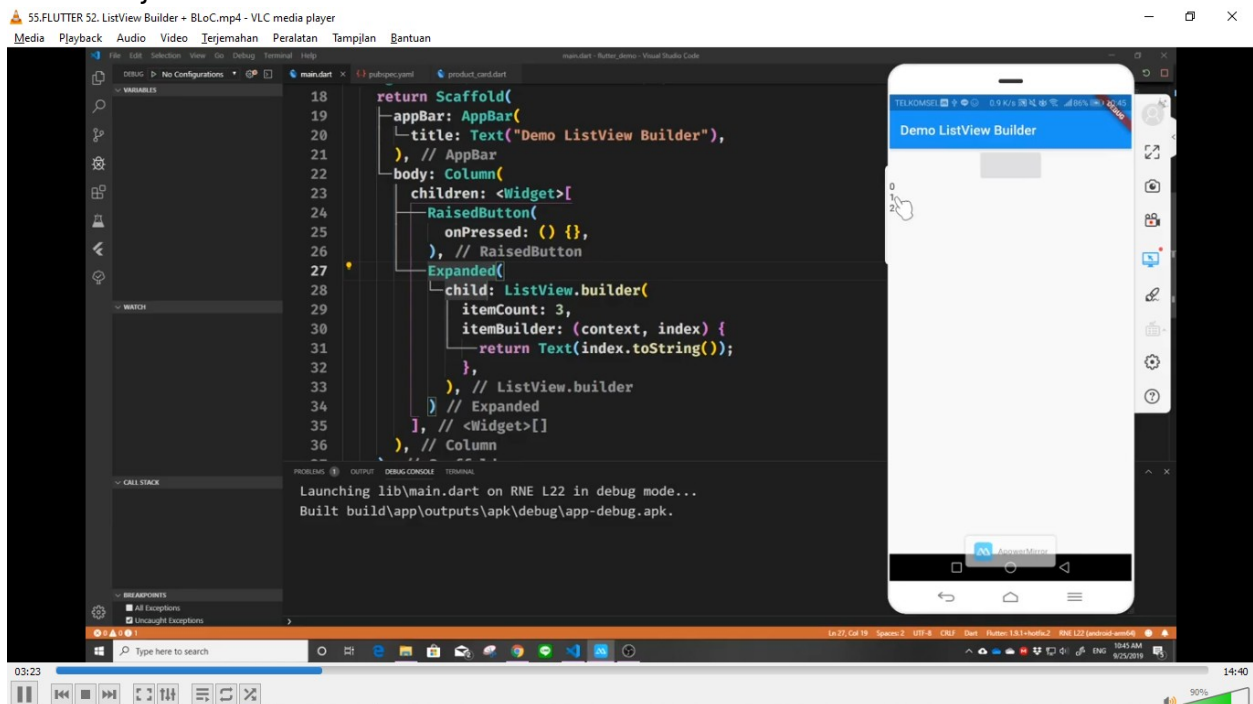
Sudah saya jelaskan
Listview buider itu sifat nya dinamis
Ada 2 hal yg perlu diisi yaitu itemcount yaitu jumlah item nya
Yg kedua
itemBuilder fungsi untuk membuat item seperti apa
supya lebih jelas maka kita praktek kan
pertama seperti biasa buat stl
materialApp
home nya mainPage
bawanya buat mainPage stl



Disini saya akan buat satu buah tombol ketika satu buah tombol itu di tekan
Dia akan membuat product card dgn jumlah random didalm subah listView builder
Pertama buat RaisedButton
Lalu dibawah buat
listView.builder
untuk percobaan kita buat
itemcount jumlah nya 3
dan itemBuilder itu masukan context dan index
di sini dia minta 2 buah paramter context dan satu buah index
nanti dia akan panggil fungsi ini 3 x
akan return text isinya index.toString buat ini saja



Lalu kita expanded kita bungkus listview
 Lalu kita jalankan



Maka dibuat kan seperti ini maka sama saja
 Gk dinamis dibuat 3 maka jalankan 3
 Kita stop dulu
 Sekarang kita buat satu buah bloc
 Bloc ini masukan angka dia akan kembalikan sejumlah product sebnk angka
 dimasukan terbut

Jadi

Kalau masukan 3 maka akan banyak produk 3 buah

Seakrang kita lanjut

Disini kita buat class ProductBloc lalu extends bloc

55.FLUTTER 52. ListView Builder + Bloc.mp4 - VLC media player

Media Playback Audio Video Terjemahan Peralatan Tampilan Bantuan

```
28 child: ListView.builder(  
29   itemCount: 3,  
30   itemBuilder: (context, index) {  
31     return Text(index.toString());  
32   },  
33 ), // ListView.builder  
34 ) // Expanded  
35 ], // <Widget>[]  
36 ), // Column  
37 ); // Scaffold  
38 }  
39 }  
40 }  
41 class ProductBloc extends Bloc {  
42 }  
43 }
```

Lauching lib\main.dart on RNE L22 in debug mode...
Built build\app\outputs\apk\debug\app-debug.apk.
Application finished.
Exited (sigterm)

04:18 14:40

Lalu import dahulu

55.FLUTTER 52. ListView Builder + Bloc.mp4 - VLC media player

Media Playback Audio Video Terjemahan Peralatan Tampilan Bantuan

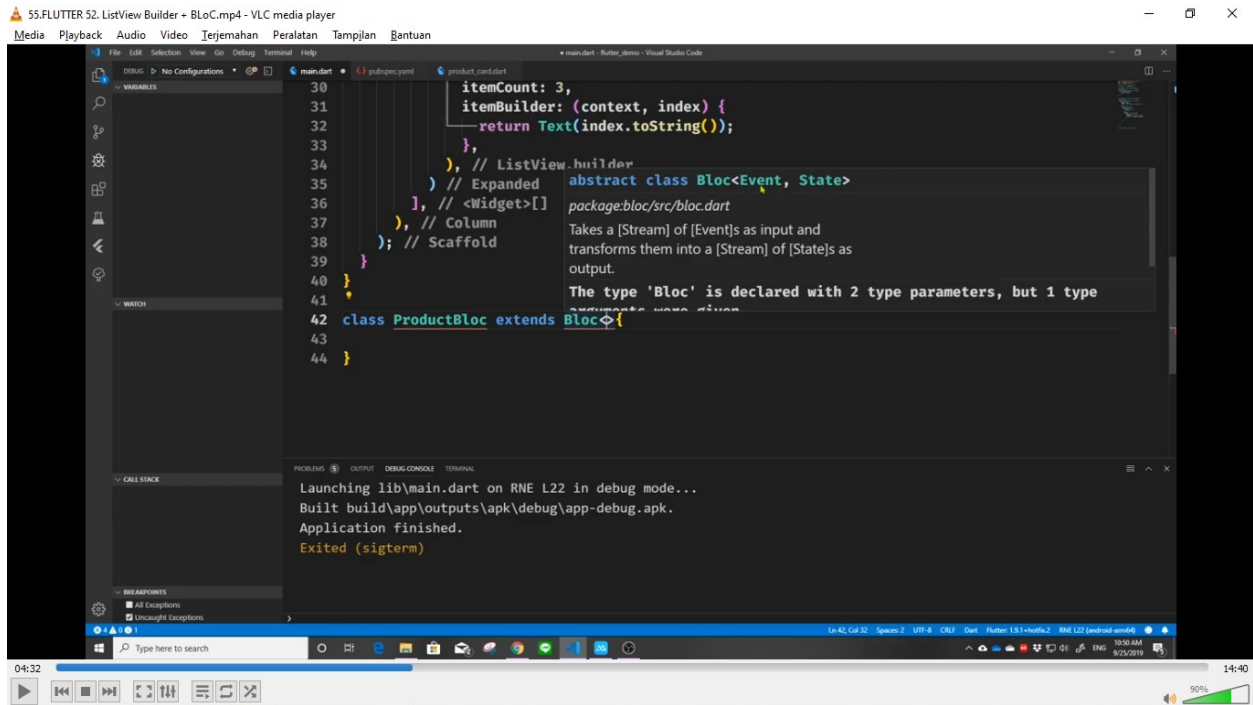
```
1 import 'package:flutter/material.dart';  
2 import 'product_card.dart';  
3 import 'package:bloc/bloc.dart';  
4  
5 void main() => runApp(MyApp());  
6  
7 class MyApp extends StatelessWidget {  
8   @override  
9   Widget build(BuildContext context) {  
10     return MaterialApp(  
11       home: MainPage(),  
12     ); // MaterialApp  
13   }  
14 }  
15  
16 class MainPage extends StatelessWidget {  
17   @override  
18   Widget build(BuildContext context) {  
19     return Scaffold(  
20       appBar: AppBar(  
21         title: Text('Product List'),  
22       ),  
23       body: ProductBloc(),  
24     );  
25   }  
26 }
```

Lauching lib\main.dart on RNE L22 in debug mode...
Built build\app\outputs\apk\debug\app-debug.apk.
Application finished.
Exited (sigterm)

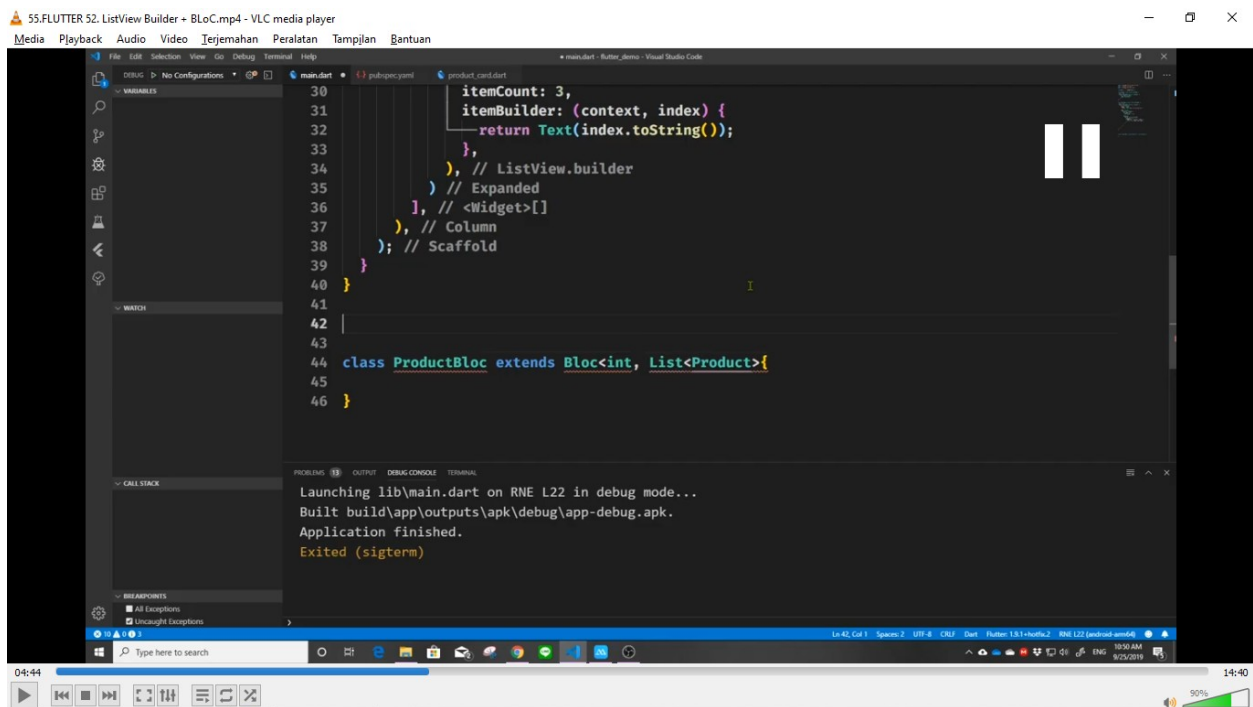
04:22 14:40

Bawah lagi

Bloc ini akan minta masukan apa keluaran apa

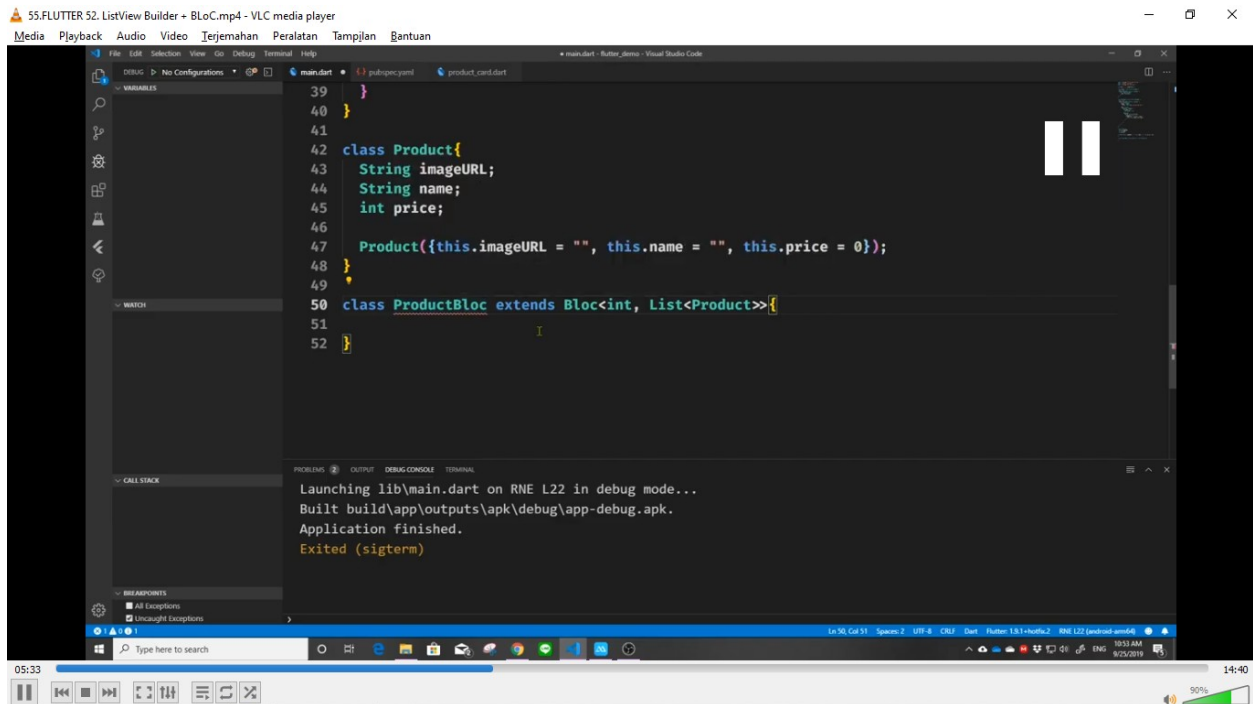


Masukan itu akan lalu kemablika itu list<Product>

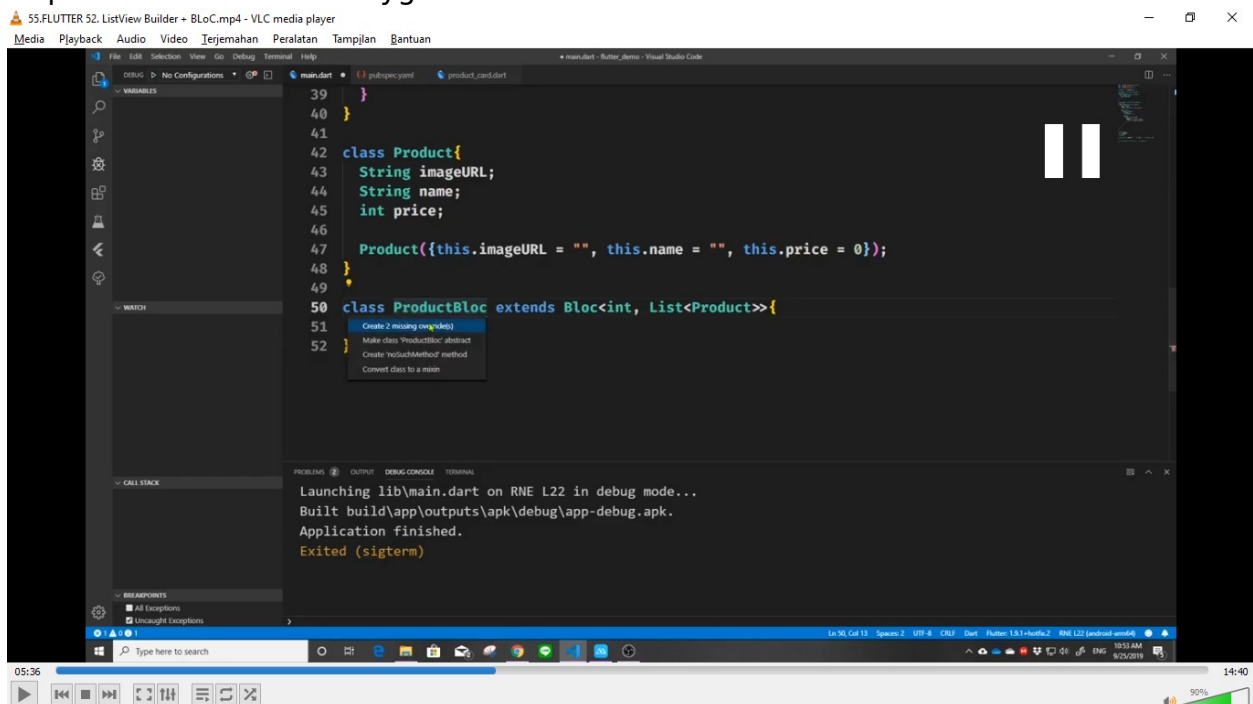


Si product kita belum buat maka buat dulu
Class product {
Dia punya string imageUrl gambarnya
Lalu punya nama product nya lalu punya harganya
Kita buat contructornya
Isi defaultnya ksoong

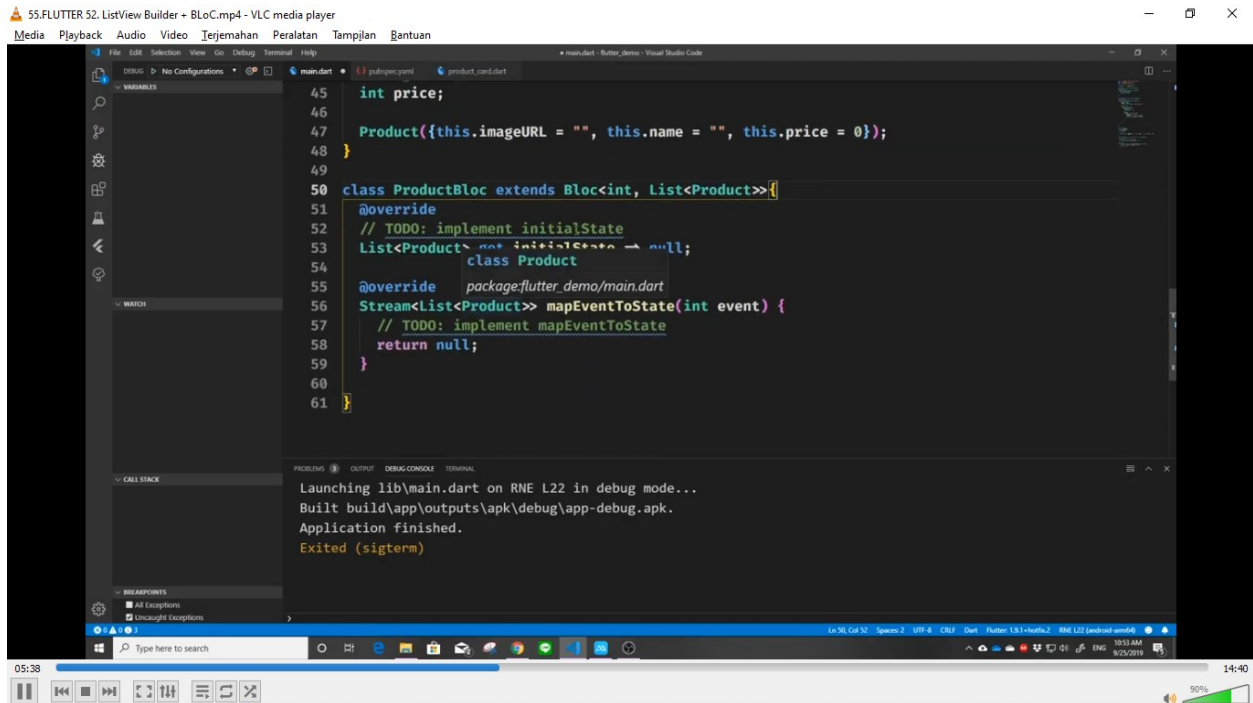
Kalau sudah disni kurang >



Di productbloc disini ada yg harus di overider

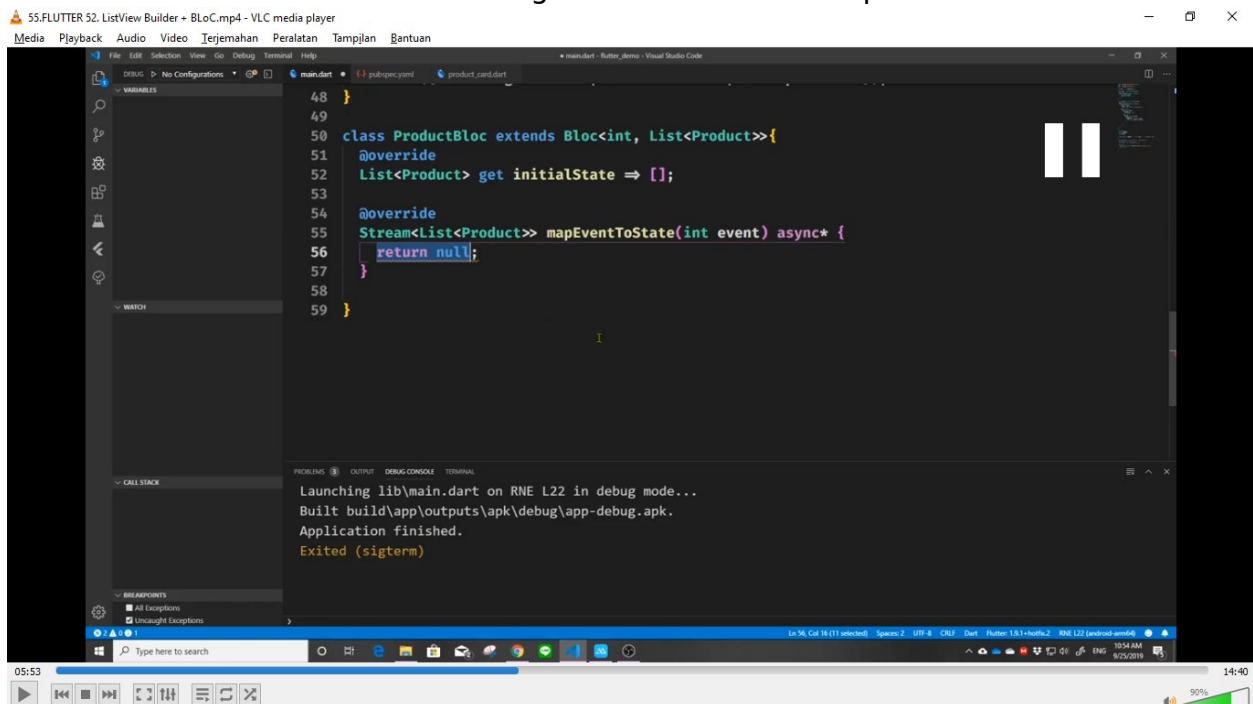


Ada 2 buah



Initial statenya list kosong

Lalu kalau miasalkan dimasukan angka dikembalikan itu apa



Maka kita buat satu buah list

Lalu kita tambhkan product.add disini

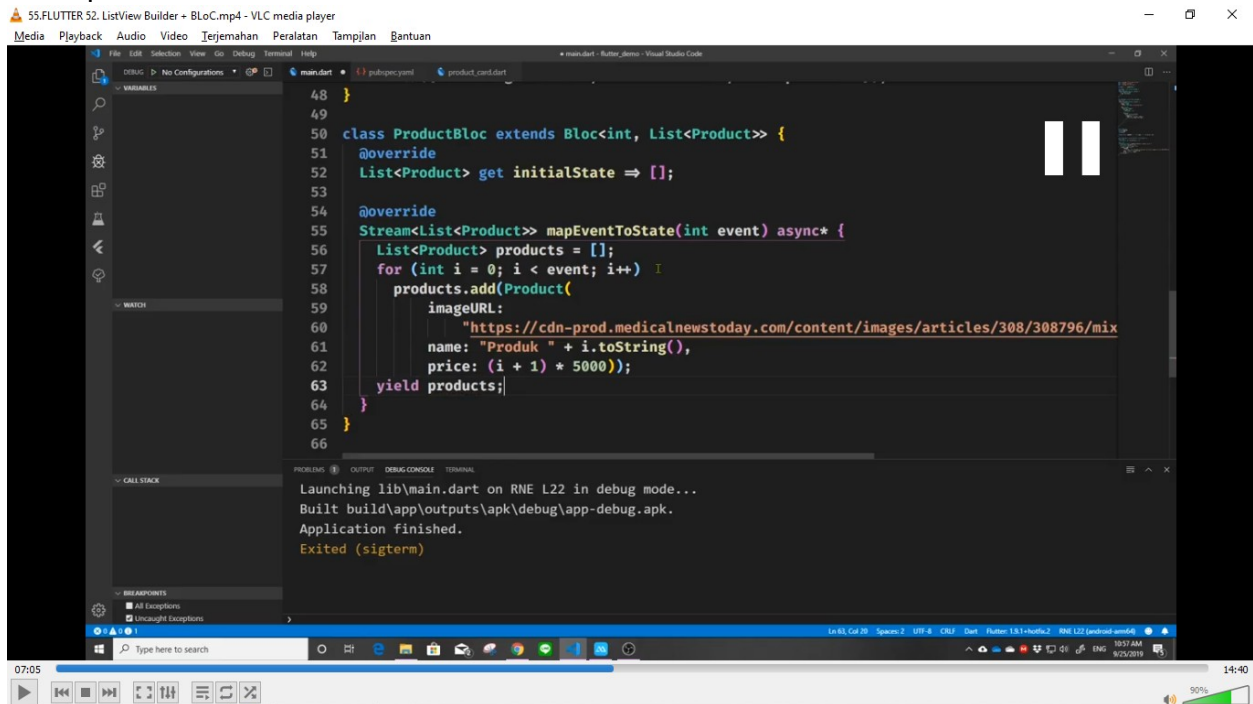
Product imageUrl itu untuk sementara static saja gambar diambil internet

, dan ada name nya lalu price nya i + 1 karena mulai dari 0 lalu * 5000

Kalau masuk nya 3 dia akan berharga 5 ribu 10 15

Setelah buat produk kita kembalikan

Yield product



The screenshot shows the VS Code editor with the `product_card.dart` file open. The code defines the `ProductBloc` class, which extends `Bloc<int, List<Product>>`. It implements `get initialState` to return an empty list and `mapEventToState` to process a list of product events. The `mapEventToState` method uses `yield` to return a stream of product lists. The debug console shows the application running successfully on RNE L22.

```
48 }
49
50 class ProductBloc extends Bloc<int, List<Product>> {
51   @override
52   List<Product> get initialState => [];
53
54   @override
55   Stream<List<Product>> mapEventToState(int event) async* {
56     List<Product> products = [];
57     for (int i = 0; i < event; i++) {
58       products.add(Product(
59         imageURL:
60           "https://cdn-prod.medicalnewstoday.com/content/images/articles/308/308796/mix",
61         name: "Produk " + i.toString(),
62         price: (i + 1) * 5000));
63       yield products;
64     }
65   }
66 }
```

Launching lib\main.dart on RNE L22 in debug mode...
Built build\app\outputs\apk\debug\app-debug.apk.
Application finished.
Exited (sigterm)

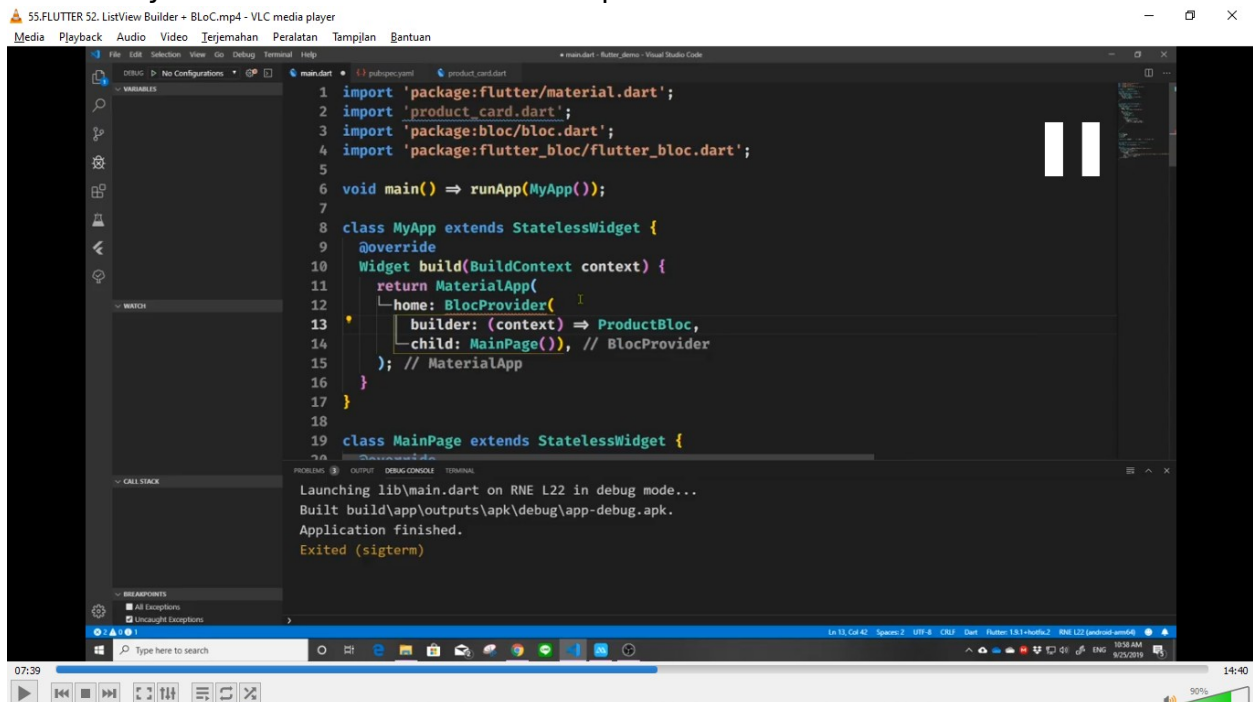
Sekarang ke atas

Disni kita import

Flutter bloc

Di mainPage kita bungkus refactor dgn widget ganti blocProvider

Didalamnya builder context kembalikan productBloc



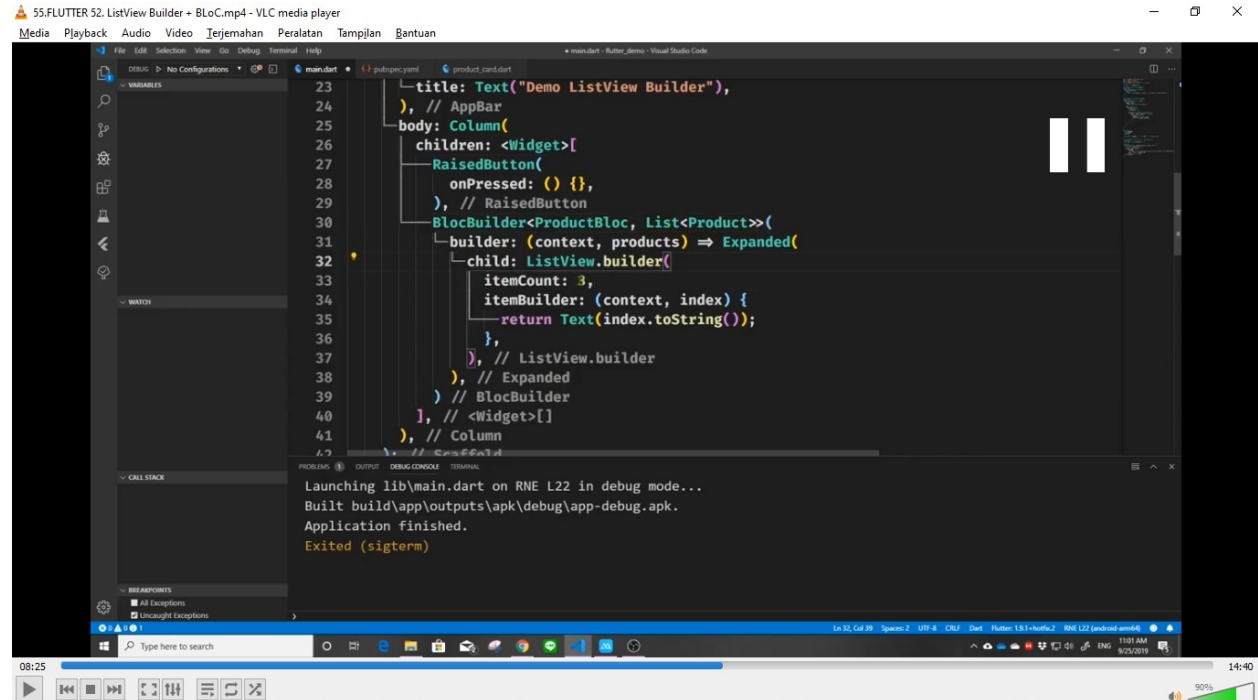
The screenshot shows the VS Code editor with the `main.dart` file open. The code imports the necessary packages for Flutter, Bloc, and Flutter Bloc. It defines the `MyApp` class, which extends `StatelessWidget`, and the `MainPage` class, which also extends `StatelessWidget`. The `MyApp` class uses `BlocProvider` to provide the `ProductBloc` to the `MainPage`. The debug console shows the application running successfully on RNE L22.

```
1 import 'package:flutter/material.dart';
2 import 'product_card.dart';
3 import 'package:bloc/bloc.dart';
4 import 'package:flutter_bloc/flutter_bloc.dart';
5
6 void main() => runApp(MyApp());
7
8 class MyApp extends StatelessWidget {
9   @override
10  Widget build(BuildContext context) {
11    return MaterialApp(
12      home: BlocProvider(
13        builder: (context) => ProductBloc,
14        child: MainPage(), // BlocProvider
15      ); // MaterialApp
16    );
17  }
18
19 class MainPage extends StatelessWidget {
20   // ...
21 }
```

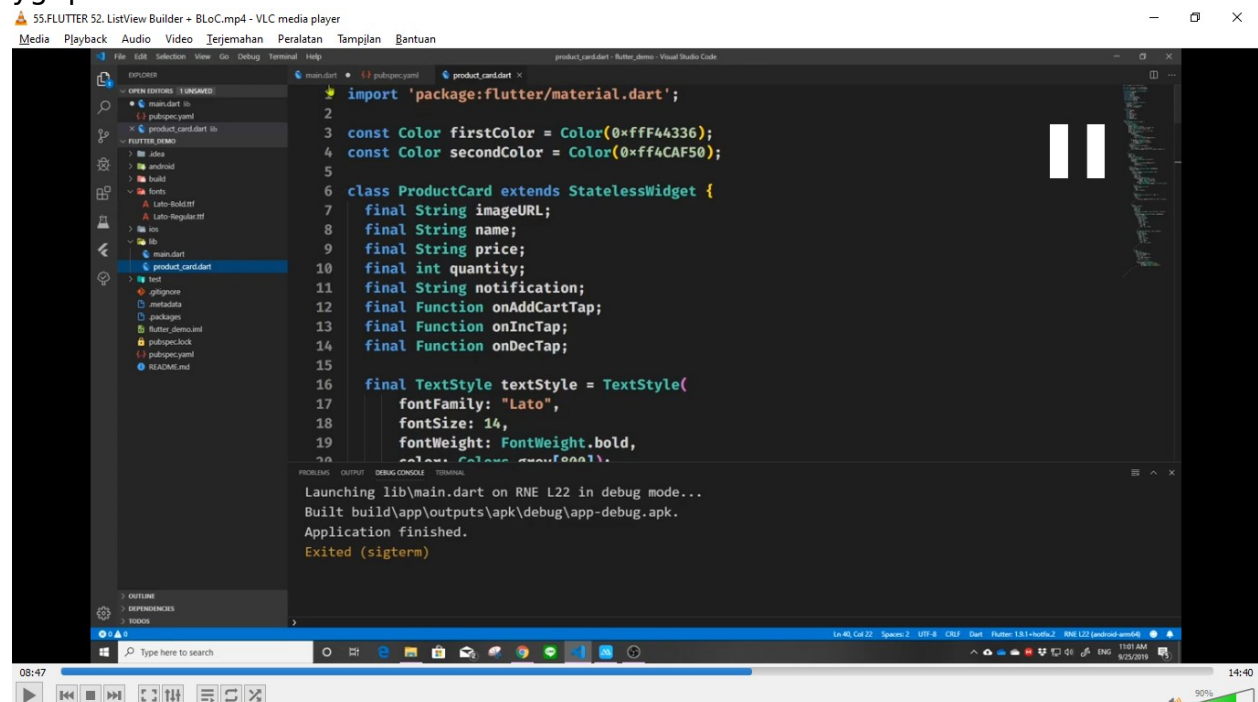
Launching lib\main.dart on RNE L22 in debug mode...
Built build\app\outputs\apk\debug\app-debug.apk.
Application finished.
Exited (sigterm)

Lalu di expanded kita bungkus lagi berikan blocBuilder kita kasih tau tipenya ProductBloc

Lalu di builder dia akan kembalikan context , lalu dia akan state nya kita product Disney list belum ditulis <product>>



Jadi disini
itemCount adalah products.length panjang nya listProduct
lalu disini akan kembalikan productCard
ygf pernah dibuat



Product card butuh img url nya kita ambil product[index].imageUrl,
Dan ada name nya

Dan ada price nya juga sama
Dia dalam bentuk string maka toString
Lalu yg lain nya kita kosongi saja

55.FLUTTER 52. ListView Builder + Bloc.mp4 - VLC media player

Media Playback Audio Video Terjemahan Peralatan Tampilan Bantuan

```
main.dart 24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
), // AppBar  
body: Column(  
  children: <Widget>[  
    RaisedButton(  
      onPressed: () {},  
    ), // RaisedButton  
    BlocBuilder<ProductBloc, List<Product>>(  
      builder: (context, products) => Expanded(  
        child: ListView.builder(  
          itemCount: products.length,  
          itemBuilder: (context, index) {  
            return ProductCard(  
              imageURL: products[index].imageURL,  
              name: products[index].name,  
              price: products[index].price.toString(),  
              onPressed: () {},  
              onDecTap: () {},  
              onIncTap: () {},  
            ); // ProductCard  
          }  
        )  
      )  
    ],  
  ),  
);
```

Launching lib\main.dart on RNE L22 in debug mode...
Built build\app\outputs\apk\debug\app-debug.apk.
Application finished.
Exited (sigterm)

Lalu untuk jumlah nya sendiri kita pasang disini
ProductBloc bloc = BlocProvider.of<

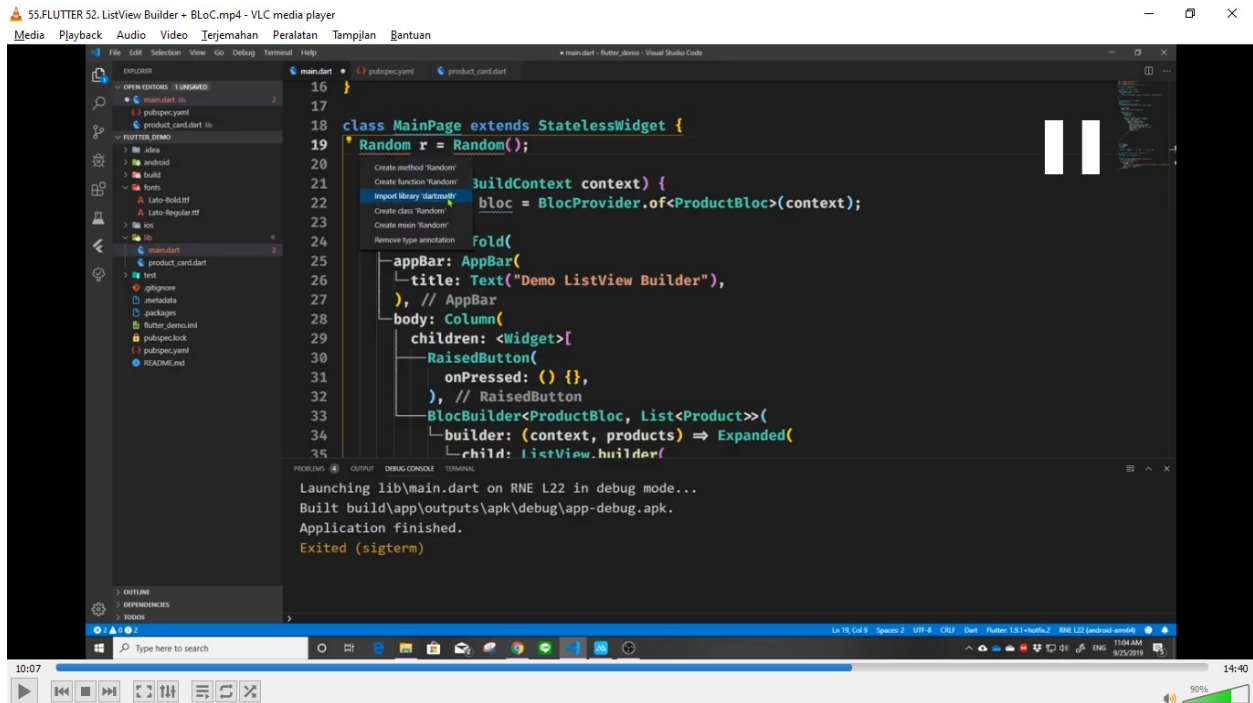
55.FLUTTER 52. ListView Builder + Bloc.mp4 - VLC media player

Media Playback Audio Video Terjemahan Peralatan Tampilan Bantuan

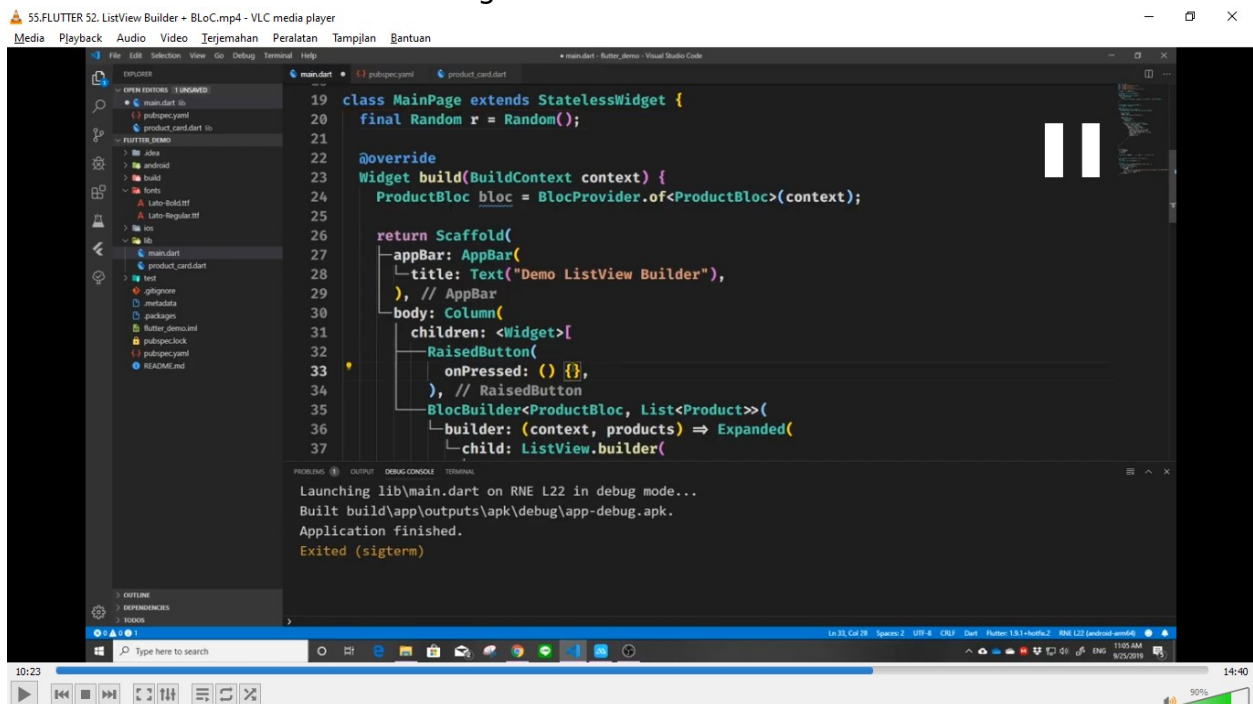
```
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
class MainPage extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    ProductBloc bloc = BlocProvider.of<ProductBloc>(context);  
  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Demo ListView Builder"),  
      ), // AppBar  
      body: Column(  
        children: <Widget>[  
          RaisedButton(  
            onPressed: () {},  
          ), // RaisedButton  
          BlocBuilder<ProductBloc, List<Product>>(  
            builder: (context, products) => Expanded(  
              child: ListView.builder(  
                itemCount: products.length,  
              )  
            )  
          ],  
        ),  
      ),  
    );  
  }  
}
```

Launching lib\main.dart on RNE L22 in debug mode...
Built build\app\outputs\apk\debug\app-debug.apk.
Application finished.
Exited (sigterm)

Lalu di mainPage buat satu buah random r = Random
Si random ini ada library maka kita import saja

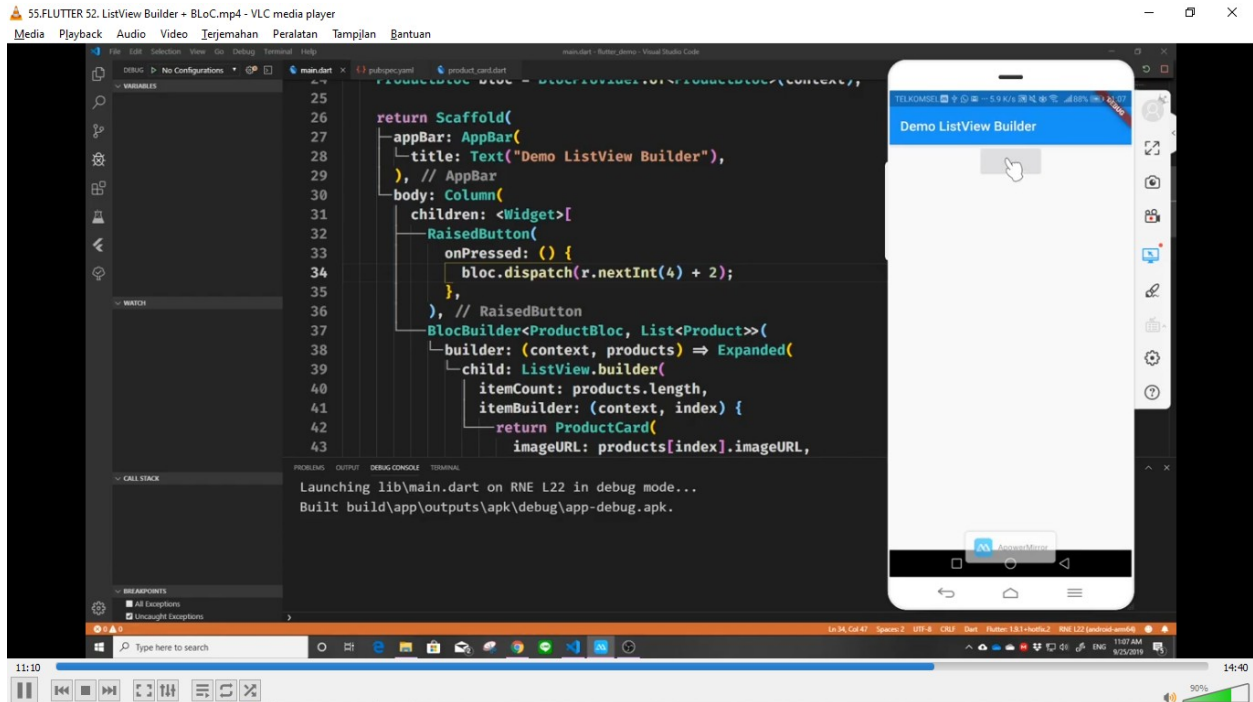


Fungsi membuat bilangan random
 Kasih final karena dia stales widget

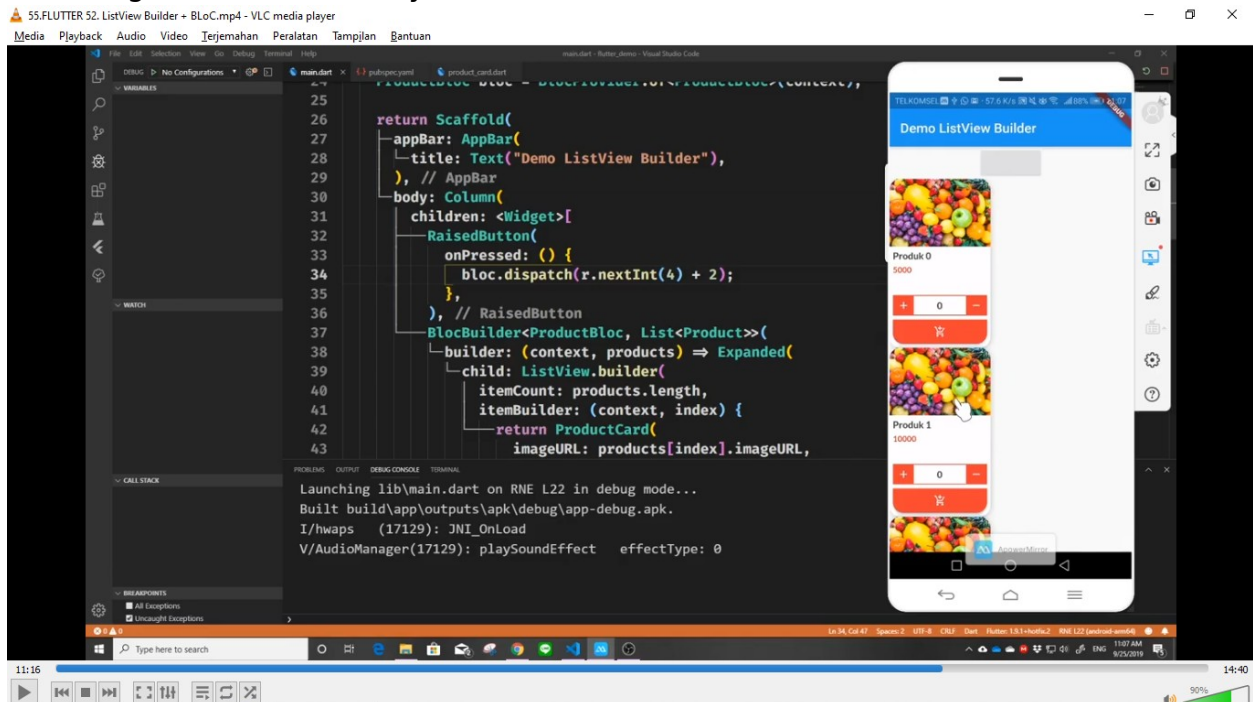


Jadi disini ketika dipencet si tombol dia akan mencipta an bilangan random
 Misalkan
 Kita random 2 atau 5 buah
 Jadi disini di onprased
 Pangil bloc.dispatch() lalu masukan angka random r.nextInt lalu kita buat max
 nya(4)

Dia akan menghasilkan antara 0,1,2,3
4 nya gk ikutan
lalu ditambhahn angka + 2 jadi hasilnya
antara
0,1,2,3,4,5;
Kita coba

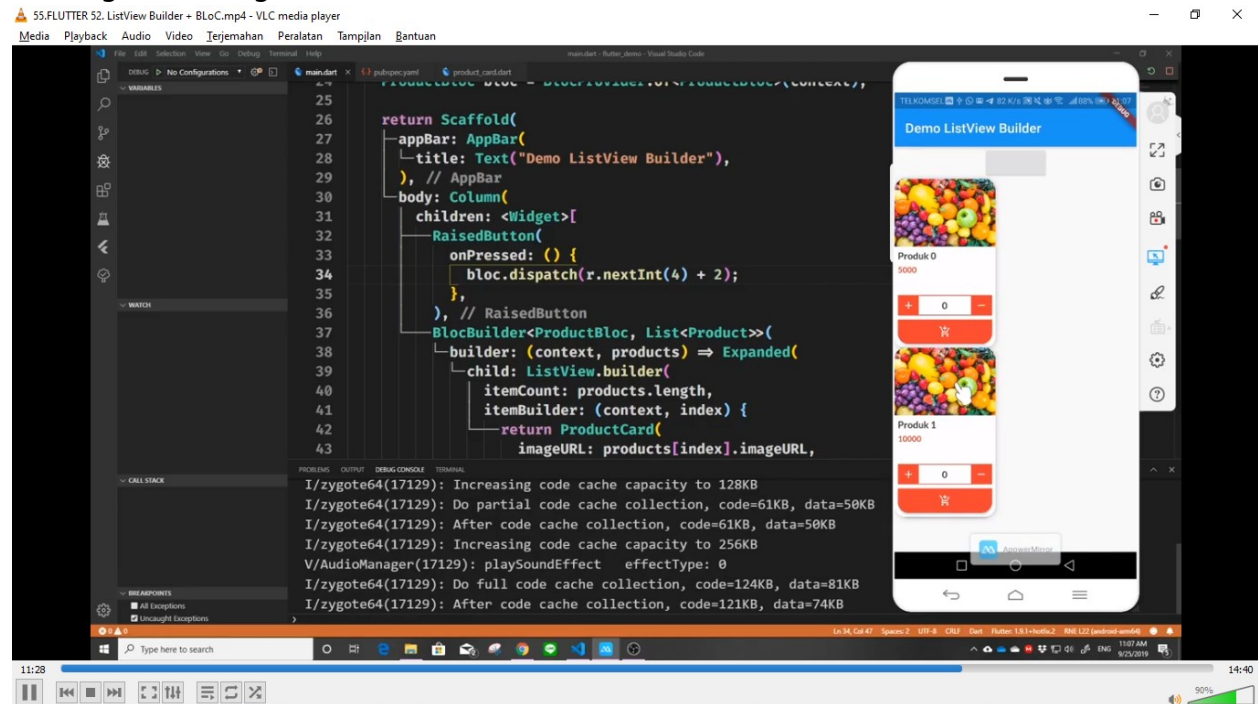


Sekarang kita klik tombol nya

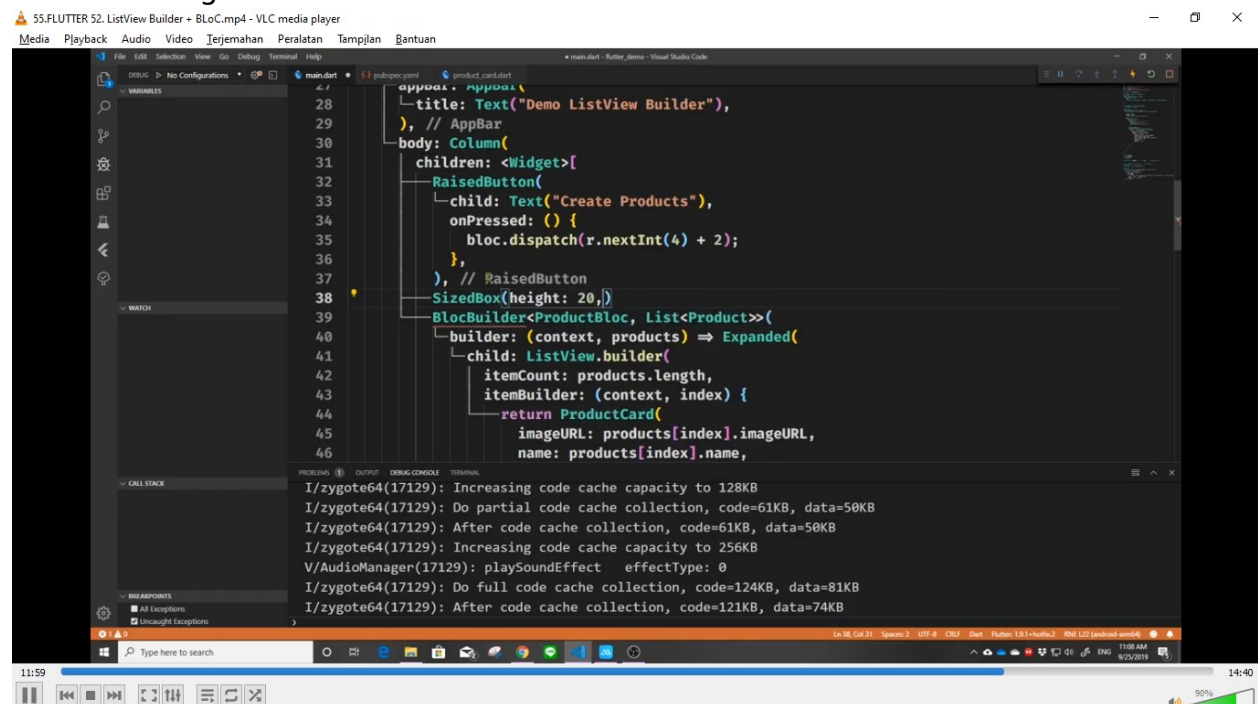


Maka dibuat 3 buah

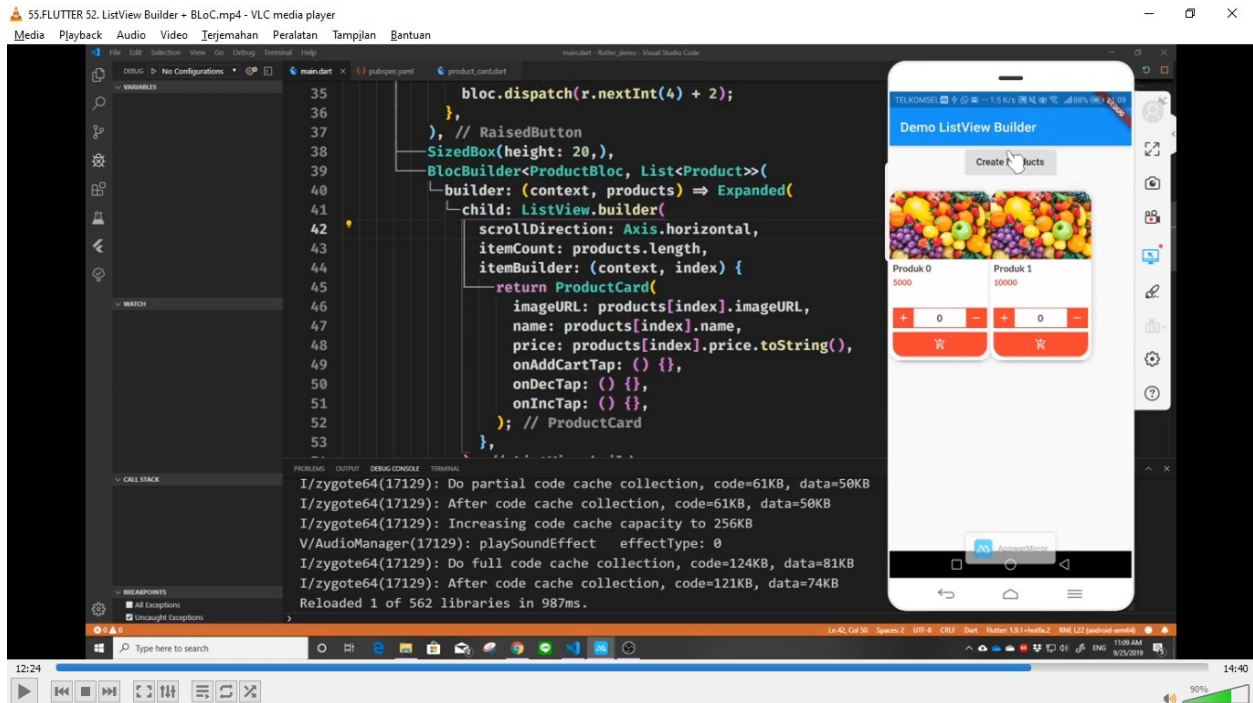
Klik lagi sekarang 2



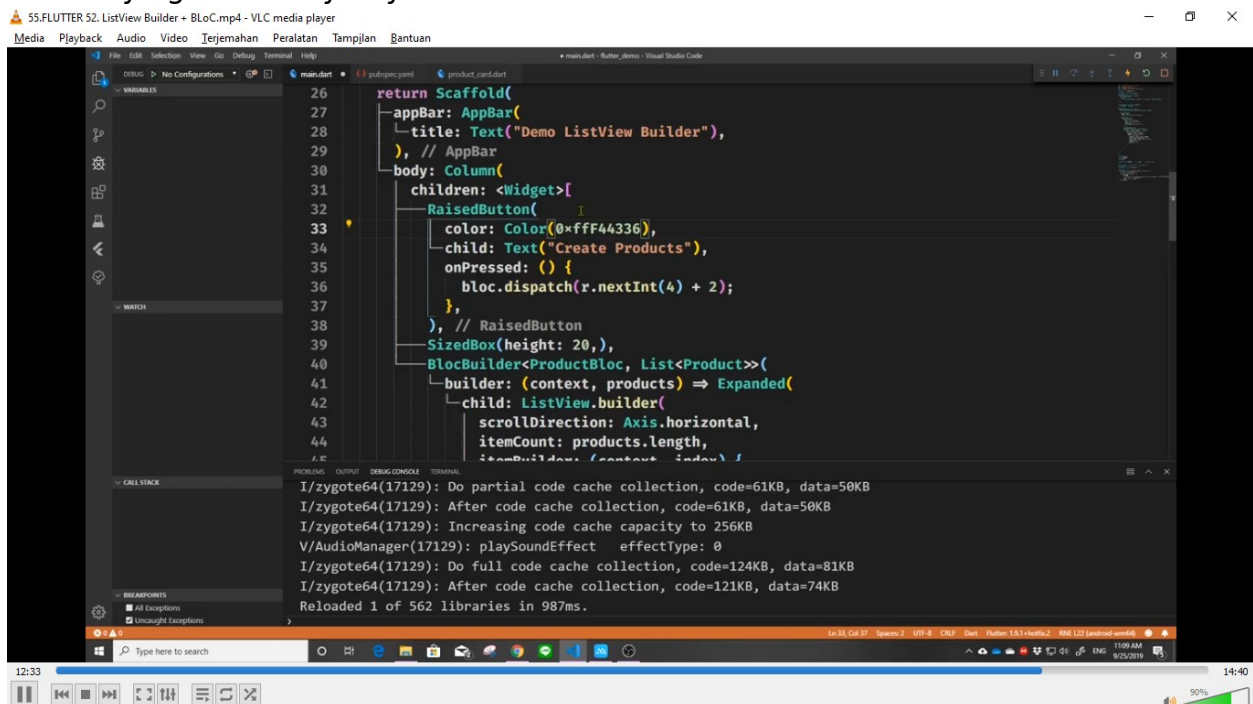
Ini biar bagus ini jelek tampilanya
Kita ubah sedikit agar menarik
Disini childnya kita buat
Setelah RaisedButton kasih jarak
SizedBox height 20



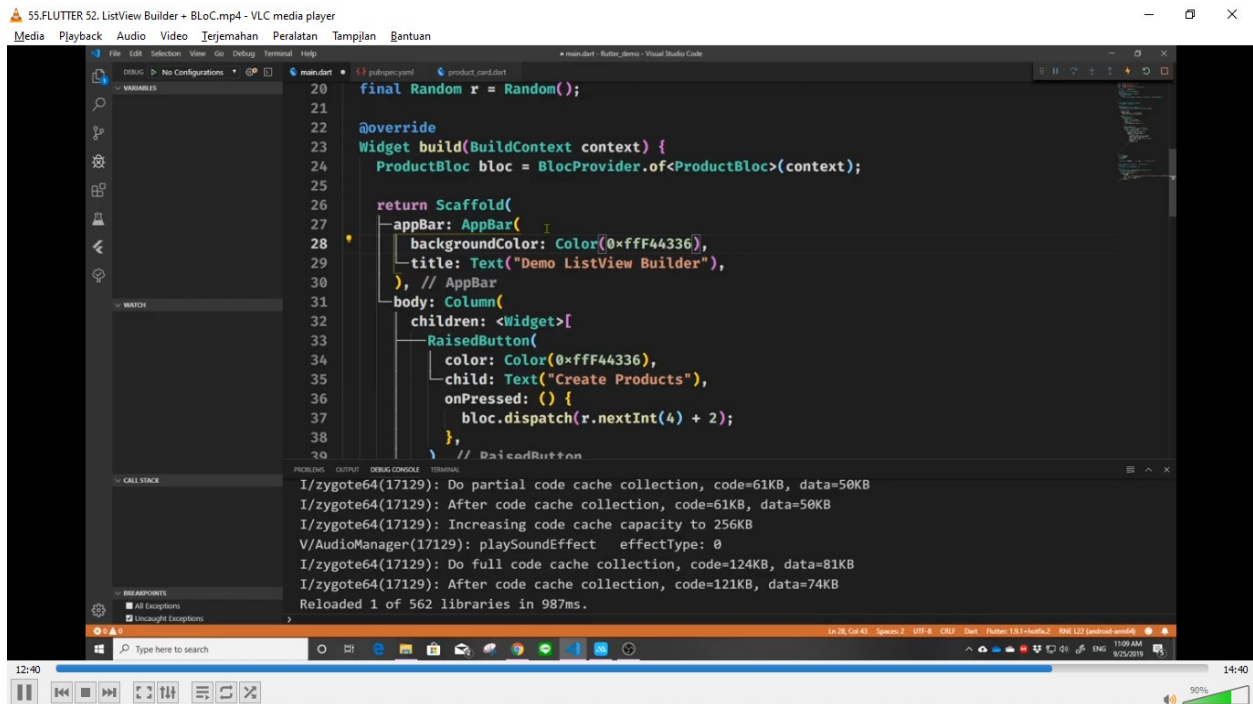
Lalu di isi ListView builder kita tentukan arah dari scrollnya
Kita buat axis horizontal mesamping



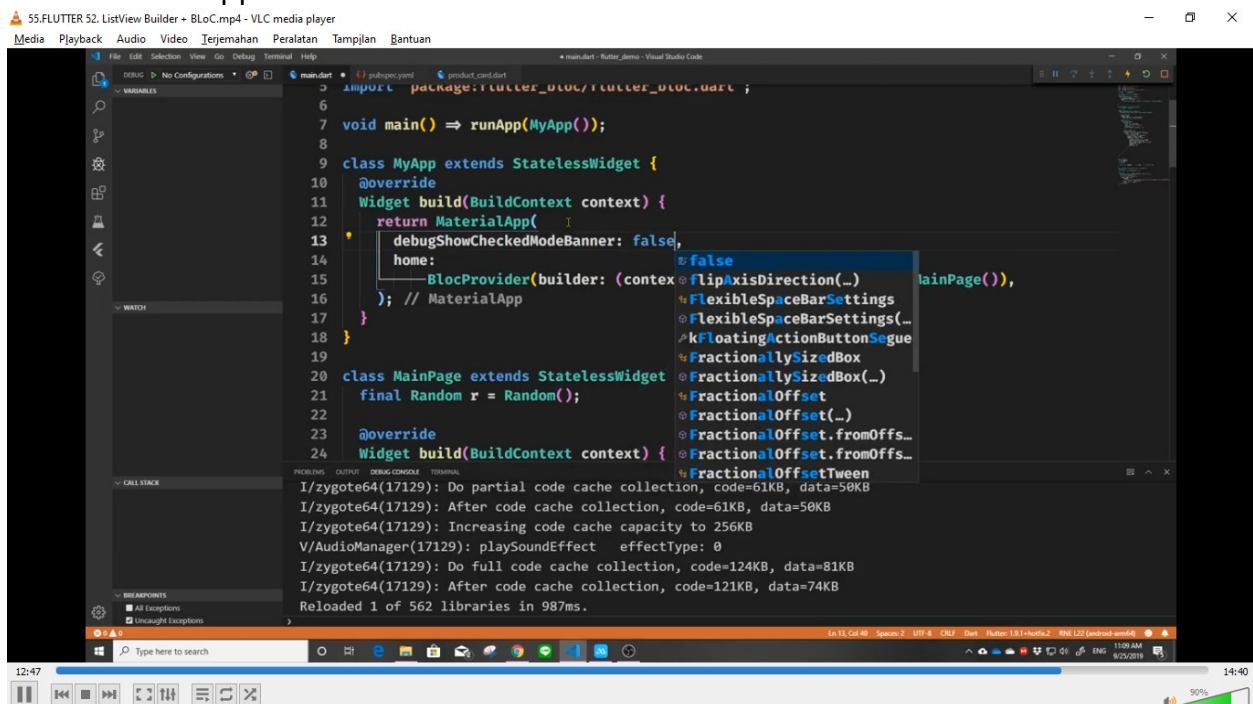
Supaya bagus kita buat tema sama warna merah
Buton nya ganti warna nya



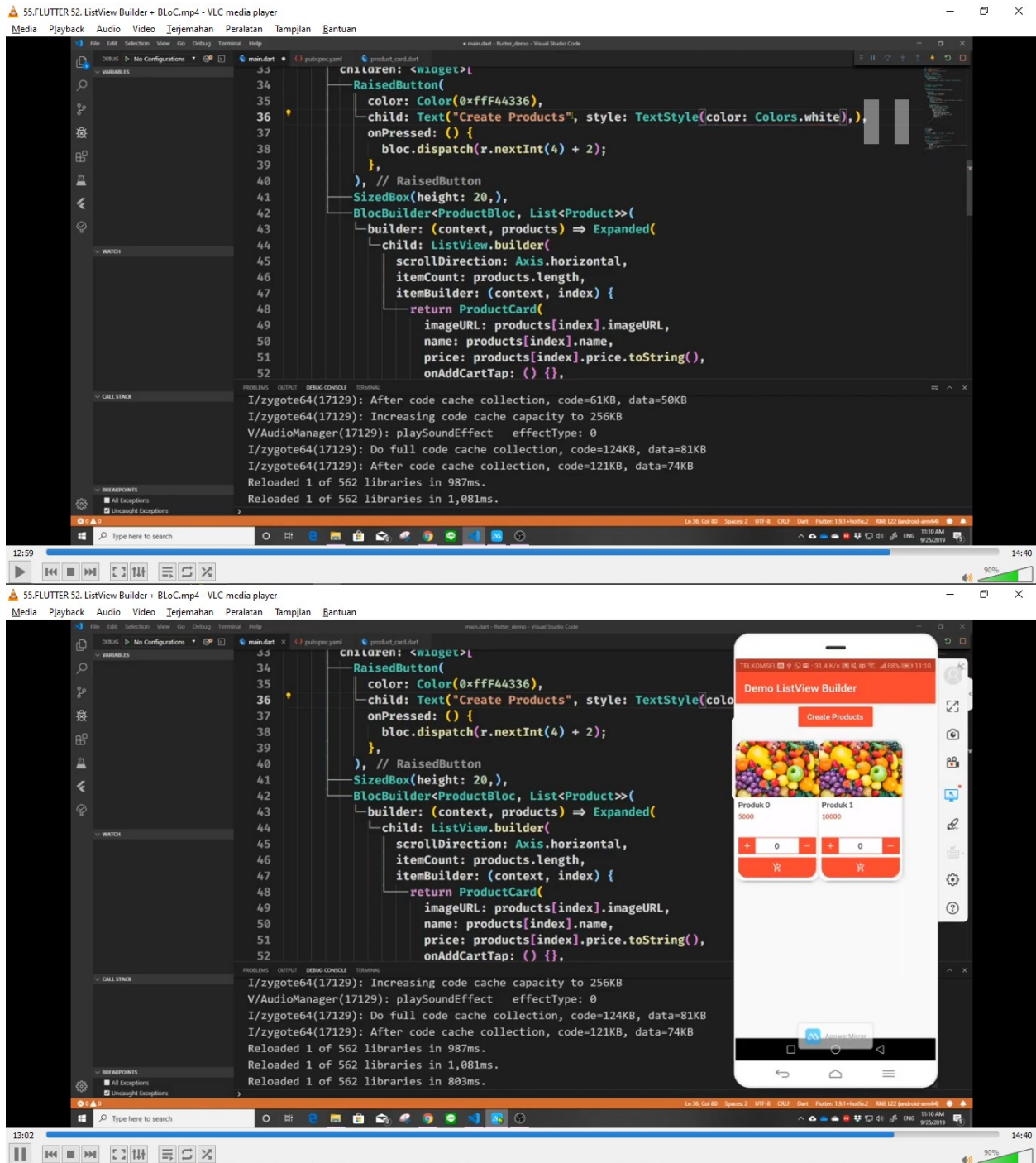
Dan appBar juga ganti



Lalu materialApp benar kita false



Tinggal kita ganti disini style nya

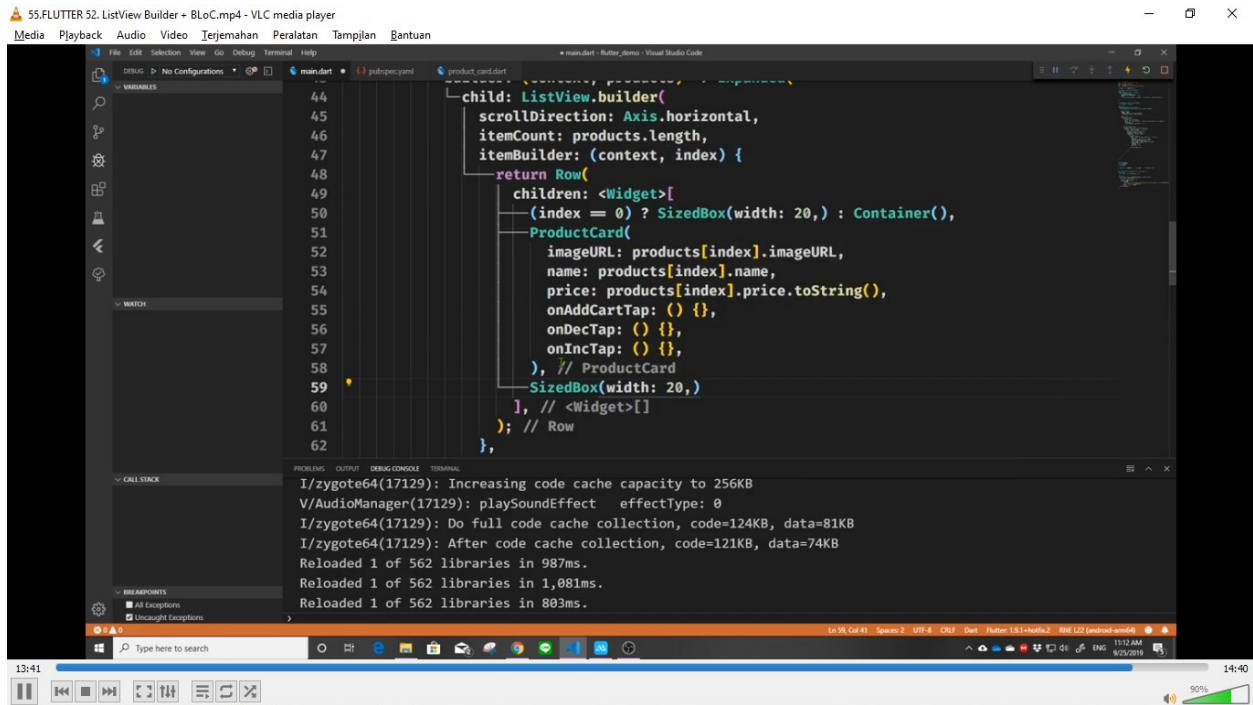


Lalu kasih jarak product card nya kita bungkus productCard dgn row sebelum productCard

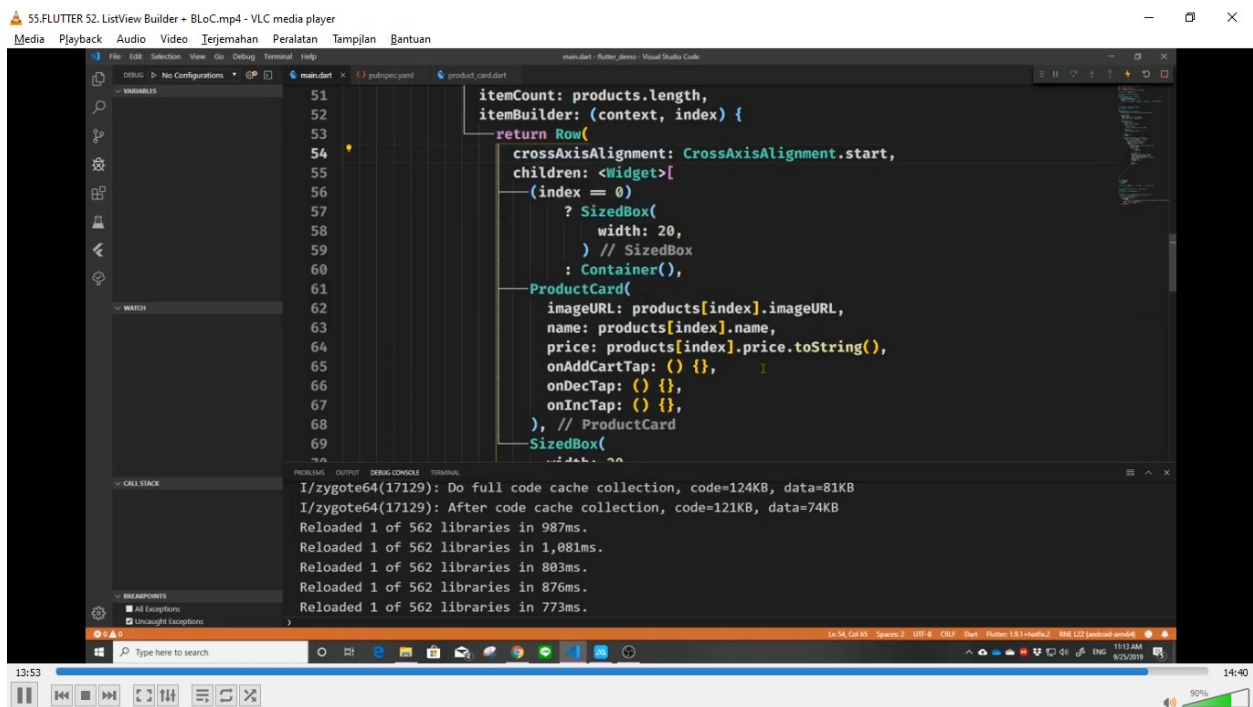
Jika ia adalah index paling pertama ==0

Buat sizedBox 20 untuk kasih jarak : kalau bukan kasih saja container

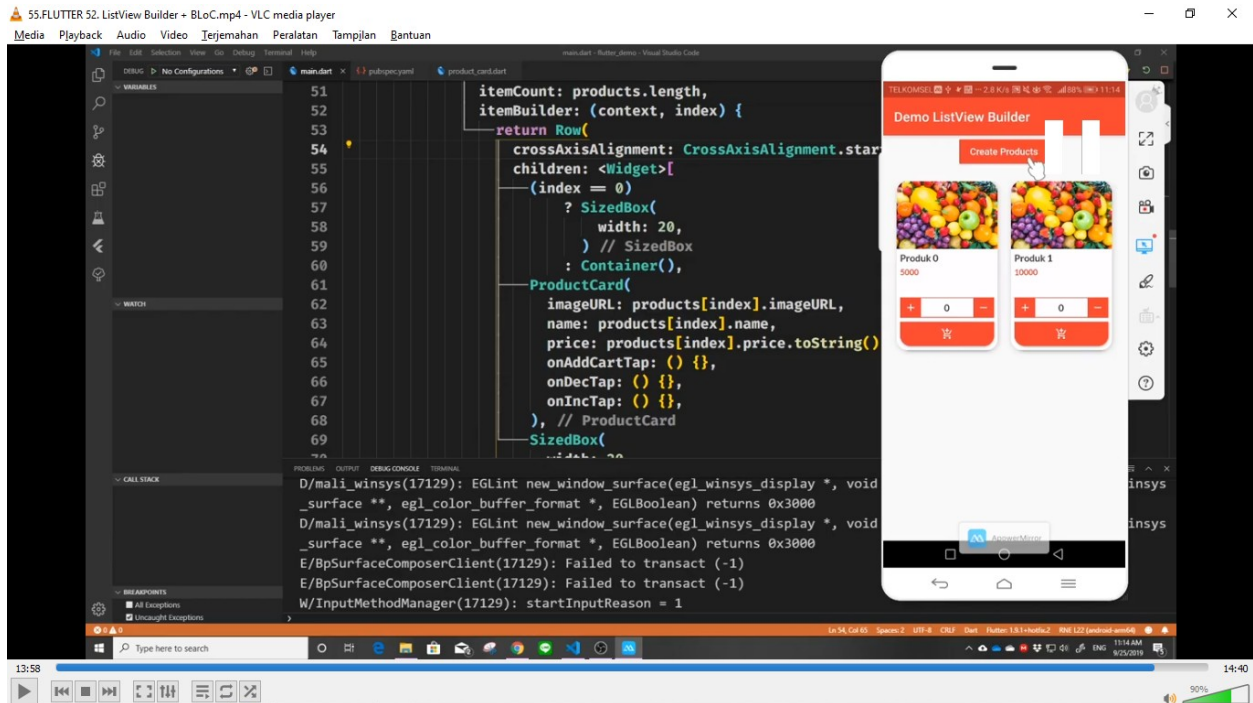
Lalu stelah procut card juga kasih jarak untuk product berikutnya



Dan disini cross kita kasih start



Kita lihat maka jadi lebih bagus



Jadi ini di klik maka bisa kesamping

Jadi ini kira gunakan listView builder gabung dgn bloc